LISTING OF THE CLAIMS

The following listing, if entered, replaces all prior versions of the claims in the present application.

1. **(Previously Presented)** A method of managing network communication comprising:

establishing a first transmission control protocol ("TCP") connection with a first network element, wherein

said first TCP connection is between said first network element and a second network element,

said first TCP connection is intended to be established with a third network element, and

said first network element comprises a transmit buffer and a receive buffer;

initiating a second TCP connection between said first network element and said third network element;

establishing communications between said second and said third network elements via said first network element;

determining that said second network element requires additional data via said first TCP connection, wherein

said determining is performed by said first network element without said additional data being requested by said second network element, and

said determining comprises detecting that acknowledged data is being removed from said transmit buffer for said first TCP connection, wherein

said detecting comprises detecting an acknowledgement sent via said first TCP connection by said second network element, and

removal of said acknowledged data frees space in said transmit buffer;

in response to said determining, pushing said additional data from said receive buffer for said second TCP connection to said transmit buffer; and

transferring said additional data from said transmit buffer to said second network element without said additional data being requested by said second network element.

2. (Original) The method of claim 1, wherein said second network element initiates said first TCP connection for said third network element.

3. (Original) The method of claim 1, wherein said communications between said second and said third network elements are established using said first and said second TCP connections.

4. (Previously Presented) The method of claim 1, wherein said communications between said second and said third network elements form an end-to-end TCP connection.

5. (Original) The method of claim 1, wherein said first network element is a proxy server.

6. (Previously Presented) The method of claim 5, wherein a control unit of said proxy server monitors said transmit buffer.

7. (Previously Presented) The method of claim 6, wherein said control unit transfers said data between said second and said third network elements.

8. (Previously Presented) The method of claim 5, wherein said proxy server supports transparent communications between said second and said third network elements.

9. (Canceled)

10. (Canceled)

11. (Previously Presented) The method of claim 1, wherein said receive buffer is pre-allocated.

12. (Previously Presented) The method of claim 1, wherein said receive buffer is dynamically allocated.

13. (Previously Presented) The method of claim 1, wherein said transmit buffer is pre-allocated.

14. (Previously Presented) The method of claim 1, wherein said transmit buffer is dynamically allocated.

15. (Original) The method of claim 1, wherein said second network element is one of a plurality of clients.

16. (Previously Presented) The method of claim 15, wherein one of a plurality of applications on said client initiates said first TCP connection for said client.

17. (Original) The method of claim 1, wherein said third network element is one of a plurality of servers.

18. (Previously Presented) The method of claim 17, wherein a data switching unit of said proxy server determines which one of said plurality of servers to use for said second TCP connection.

19. (Original) The method of claim 1, further comprising:
monitoring said first TCP connection.

20. (Previously Presented) The method of claim 19, further comprising:
receiving a request for data from said second network element; and
determining whether said request requires said second TCP connection with one of said plurality of servers.

21. (Original) The method of claim 20, wherein data switching unit receives said request for data via said control unit.

22. (Original) The method of claim 20, wherein said determining of said second TCP connection is done by said data switching unit.

23. (Previously Presented) The method of claim 20, further comprising:
if said request does not require said second TCP connection with one of said plurality of servers,
servicing said request for data, and
closing said connection with said second network element.

24. (Previously Presented) The method of claim 23, wherein said request for data is served by passing data from said data switching unit to said control unit for transmission to an application on said second network element.

25. (Original) The method of claim 23, further comprising:
if said request requires said second TCP connection with one of said plurality of servers,
selecting a first server from said plurality of servers, and
initiating said second TCP connection with said first server.

26. (Original) The method of claim 25, wherein said application requests said end-to-end TCP connection with said first server.

27. (Previously Presented) The method of claim 25, further comprising:

receiving said data on said second TCP connection from said first server;

storing said data in said receive buffer of said second TCP connection.

28. **(Cancelled)**

29. **(Cancelled)**

30. **(Currently Amended)** The method of claim 1, further comprising:

closing said first TCP connection with said client **in response to receiving a request for closing said connection from said data switching unit**.

31. **(Currently Amended)** The method of claim 30, ~~**wherein said closing of said connection is done by said control unit upon a receiving a request for closing said connection from said data switching unit**~~ **further comprising:**

performing a two-stage operation for releasing a control memory entry for said connection, wherein said control memory entry comprises a connection block entry and a flow control entry, and said two-stage operation comprises:

releasing said connection block entry prior to receiving said request for closing said connection; and

releasing said flow control entry in response to receiving said request for closing said connection.

32. **(Previously Presented)** A network device configured to:

establish a first transmission control protocol ("TCP") connection with a first network element, wherein

said first TCP connection is between said first network element and a second network element,

said first TCP connection is intended to be established with a third network element, and

said first network element comprises a transmit buffer and a receive buffer;

initiate a second TCP connection between said first network element and said third network element;

establish communications between said second and said third network elements via said first network element;

determine that said second network element requires additional data via said first TCP connection, wherein

said determination is performed by said first network element without said additional data being requested by said second network element, and

said determination comprises a detection that acknowledged data is being removed from said transmit buffer for said first TCP connection, wherein

said detection comprises detecting an acknowledgement sent via said first TCP connection by said second network element, and

removal of said acknowledged data frees space in said transmit buffer;

in response to said determination, push said additional data from said receive buffer for said second TCP connection to said transmit buffer; and

transfer said additional data from said transmit buffer to said second network element without said additional data being requested by said second network element.

33. (Original) The network device of claim 32, wherein said second network element initiates said first TCP connection for said third network element.

34. (Original) The network device of claim 32, wherein said communications between said second and said third network elements are established using said first and said second TCP connections.

35. (Previously Presented) The network device of claim 32, wherein said communications between said second and said third network elements form an end-to-end TCP connection.

36. (Original) The network device of claim 32, wherein said first network element is a proxy server.

37. (Previously Presented) The network device of claim 36, wherein a control unit of said proxy server monitors said transmit buffer.

38. (Previously Presented) The network device of claim 37, wherein said control unit transfers said data between said second and said third network elements.

39. (Previously Presented) The network device of claim 36, wherein said proxy server supports transparent communications between said second and said third network elements.

40. (Canceled)

41. (Canceled)

42. (Previously Presented) The network device of claim 32, wherein said receive buffer is pre-allocated.

43. (Previously Presented) The network device of claim 32, wherein said receive buffer is dynamically allocated.

44. (Previously Presented) The network device of claim 32, wherein said transmit buffer is pre-allocated.

45. (Previously Presented) The network device of claim 32, wherein said transmit buffer is dynamically allocated.

46. (Original) The network device of claim 32, wherein said second network element is one of a plurality of clients.

47. (Previously Presented) The network device of claim 46, wherein one of a plurality of applications on said client initiates said first TCP connection for said client.

48. (Original) The network device of claim 32, wherein said third network element is one of a plurality of servers.

49. (Previously Presented) The network device of claim 48, wherein a data switching unit of said proxy server determines which one of said plurality of servers to use for said second TCP connection.

50. (Previously Presented) The network device of claim 32, wherein said network device is further configured to monitor said first TCP connection.

51. (Previously Presented) The network device of claim 50, wherein said network device is further configured to receive a request for data from an application on a client and determine whether said request requires said second TCP connection with one of a plurality of servers.

52. (Previously Presented) The network device of claim 51, wherein a data switching unit receives said request for data via a control unit.

53. (Previously Presented) The network device of claim 51, wherein said determining of said second TCP connection is done by a data switching unit.

54. (Previously Presented) The network device of claim 51, wherein said network device is further configured to if said request does not require said second TCP connection with one of said plurality of servers,

service said request for data, and

close said connection with said client.

55. (Original) The network device of claim 54, wherein said request for data is served by passing data from said data switching unit to said control unit for transmission to said application on said client.

56. (Previously Presented) The network device of claim 54, wherein said network device is further configured to if said request requires said second TCP connection with one of said plurality of servers,

select a first server from said plurality of servers, and

initiate said second TCP connection with said first server.

57. (Original) The network device of claim 56, wherein said application requests said end-to-end TCP connection with said first server.

58. (Previously Presented) The network device of claim 56, wherein said network device is further configured to receive said data on said second TCP connection from said first server;

store said data in said receive buffer of said second TCP connection.

59. **(Cancelled)**

60. **(Cancelled)**

61. **(Currently Amended)** The network device of claim 32, wherein said network device is further configured to close said first TCP connection with said client **in response to receiving a request for closing said connection from said data switching unit**.

62. **(Currently Amended)** The network device of claim 61, ~~wherein said closing of said connection is done by said control unit upon a receiving a request for closing said connection from said data switching unit~~ **wherein said network device is further configured to perform a two-stage operation for releasing a control memory entry for said connection, wherein said control memory entry comprises a connection block entry and a flow control entry, and said two-stage operation comprises:**

releasing said connection block entry prior to receiving said request for closing said connection; and

releasing said flow control entry in response to receiving said request for closing said connection.

63. **(Previously Presented)** A network device comprising:

means for establishing a first transmission control protocol ("TCP") connection with a first network element, wherein

said first TCP connection is between said first network element and a second network element,

said first TCP connection is intended to be established with a third network element, and

said first network elements comprises a transmit buffer and a receive buffer;

means for initiating a second TCP connection between said first network element and said third network element;

means for establishing communications between said second and said third network elements via said first network element;

means for determining that said second network element requires additional data via said first TCP connection, wherein

said determining is performed by said means for determining without said additional data being requested by said second network element, and

said determining comprises detecting that acknowledged data is being

removed from said transmit buffer for said first TCP connection, wherein

said means for detecting comprises means for detecting an acknowledgement sent via said first TCP connection by said second network element, and

removal of said acknowledged data frees space in said transmit buffer;

means for pushing said additional data from said receive buffer for said second TCP connection to said transmit buffer, in response to said determining; and

means for transferring said additional data from said transmit buffer to said second network element without said additional data being requested by said second network element.

64. (Original) The network device of claim 63, wherein said second network element initiates said first TCP connection for said third network element.

65. (Original) The network device of claim 63, wherein said communications between said second and said third network elements are established using said first and said second TCP connections.

66. (Previously Presented) The network device of claim 63, wherein said communications between said second and said third network elements form an end-to-end TCP connection.

67. (Original) The network device of claim 63, wherein said first network element is a proxy server.

68. (Previously Presented) The network device of claim 67, wherein a control unit of said proxy server monitors said transmit buffer.

69. (Previously Presented) The network device of claim 68, wherein a control unit transfers said data between said second and said third network elements.

70. (Previously Presented) The network device of claim 67, wherein said proxy server supports transparent communications between said second and said third network elements.

71. (Canceled)

72. (Canceled)

73. (Previously Presented) The network device of claim 63, wherein said receive buffer is pre-allocated.

74. (Previously Presented) The network device of claim 63, wherein said receive buffer is dynamically allocated.

75. (Previously Presented) The network device of claim 63, wherein said transmit buffer is pre-allocated.

76. (Previously Presented) The network device of claim 63, wherein said transmit buffer is dynamically allocated.

77. (Previously Presented) The network device of claim 63, wherein said second network element is one of a plurality of clients.

78. (Previously Presented) The network device of claim 77, wherein one of a plurality of applications on said client initiates said first TCP connection for said client.

79. (Previously Presented) The network device of claim 63, wherein said third network element is one of a plurality of servers.

80. (Previously Presented) The network device of claim 79, wherein a data switching unit of a proxy server determines which one of said plurality of servers to use for said second TCP connection.

81. (Previously Presented) The network device of claim 63, further comprising:

means for monitoring said first TCP connection.

82. (Previously Presented) The network device of claim 81, further comprising:

means for receiving a request for data from an application on a client; and

means for determining whether said request requires said second TCP connection with one of a plurality of servers.

83. (Previously Presented) The network device of claim 82, wherein a data switching unit receives said request for data via a control unit.

84. (Previously Presented) The network device of claim 82, wherein said determining of said second TCP connection is done by a data switching unit.

85. (Original) The network device of claim 82, further comprising:

means for servicing said request for data if said request does not require said second TCP connection with one of said plurality of servers; and

means for closing said connection with said client if said request does not require said second TCP connection with one of said plurality of servers.

86. (Original) The network device of claim 85, wherein said request for data is served by passing data from said data switching unit to said control unit for transmission to said application on said client.

87. (Original) The network device of claim 85, further comprising:

means for selecting a first server from said plurality of servers if said request

requires said second TCP connection with one of said plurality of servers; and

means for initiating said second TCP connection with said first server if said request requires said second TCP connection with one of said plurality of servers.

88. (Original) The network device of claim 87, wherein said application requests said end-to-end TCP connection with said first server.

89. (Previously Presented) The network device of claim 87, further comprising:

means for receiving said data on said second TCP connection from said first server;

means for storing said data in said receive buffer of said second TCP connection.

90. (Previously Presented) The network device of claim 89, further comprising:

means for requesting said additional data from said first server if said first TCP connection needs said additional data.

91. (Original) The network device of claim 90, wherein said additional data is transferred into said transmit buffer without a request for said additional data.

92. **(Currently Amended)** The network device of claim 90, further comprising:

means for closing said first TCP connection with said client **in response to receiving a request for closing said connection from said data switching unit** if said request for data from said application is served.

93. **(Currently Amended)** The network device of claim 92, ~~**wherein said closing of said connection is done by said control unit upon a receiving a request for closing said connection from said data switching unit**~~ **further**

comprising means for performing a two-stage operation for releasing a control memory entry for said connection, wherein said control memory entry comprises a connection block entry and a flow control entry, and said two-stage operation comprises:

releasing said connection block entry prior to receiving said request for closing said connection; and

releasing said flow control entry in response to receiving said request for closing said connection.

94. **(Previously Presented)** A computer program product for managing network communication, encoded in computer readable media, said program product comprising a set of instructions executable on a computer system, said set of instructions configured to:

establish a first transmission control protocol ("TCP") connection with a first network element, wherein

said first TCP connection is between said first network element and a second network element,

said first TCP connection is intended to be established with a third network element, and

said first network element comprises a transmit buffer and a receive buffer;

initiate a second TCP connection between said first network element and said third network element;

establish communications between said second and said third network elements via said first network element;

determine that said second network element requires additional data via said first TCP connection, wherein

said determining is performed by said first network element without said additional data being requested by said second network element, and

said determining comprises detect that acknowledged data is being

removed from said transmit buffer for said first TCP connection, wherein

detection that acknowledged data is being removed comprises detection of an acknowledgement sent via said first TCP connection by said second network element, and

removal of said acknowledged data frees space in said transmit buffer;

in response to said determination, push said additional data from said receive buffer for said second TCP connection to said transmit buffer; and

transfer said additional data from said transmit buffer to said second network element without said additional data being requested by said second network element.

95. (Original) The computer program product of claim 94, wherein said second network element initiates said first TCP connection for said third network element.

96. (Original) The computer program product of claim 94, wherein said communications between said second and said third network elements are established using said first and said second TCP connections.

97. (Previously Presented) The computer program product of claim 94, wherein said communications between said second and said third network elements form an end-to-end TCP connection.

98. (Original) The computer program product of claim 94, wherein said first network element is a proxy server.

99. (Previously Presented) The computer program product of claim 98, wherein a control unit of said proxy server monitors said plurality of buffers.

100. (Previously Presented) The computer program product of claim 99,

wherein said control unit transfers said data between said second and said third network elements.

101. (Previously Presented) The computer program product of claim 98, wherein said proxy server supports transparent communications between said second and said third network elements.

102. (Canceled)

103. (Canceled)

104. (Previously Presented) The computer program product of claim 94, wherein said receive buffer is pre-allocated.

105. (Previously Presented) The computer program product of claim 94, wherein said receive buffer is dynamically allocated.

106. (Previously Presented) The computer program product of claim 94, wherein said transmit buffer is pre-allocated.

107. (Previously Presented) The computer program product of claim 94, wherein said transmit buffer is dynamically allocated.

108. (Original) The computer program product of claim 94, wherein said second network element is one of a plurality of clients.

109. (Previously Presented) The computer program product of claim 108, wherein one of a plurality of applications on said client initiates said first TCP connection for said client.

110. (Original) The computer program product of claim 94, wherein said third network element is one of a plurality of servers.

111. (Previously Presented) The computer program product of claim 110, wherein a data switching unit of said proxy server determines which one of said plurality of servers to use for said second TCP connection.

112. (Original) The computer program product of claim 94, wherein said set of instructions is further configured to:
monitor said first TCP connection.

113. (Previously Presented) The computer program product of claim 112, wherein said set of instructions is further configured to:
receive a request for data from an application on a client; and
determine whether said request requires said second TCP connection with one of a plurality of servers.

114. (Previously Presented) The computer program product of claim 113, wherein a data switching unit receives said request for data via a control unit.

115. (Original) The computer program product of claim 113, wherein said determining of said second TCP connection is done by said data switching unit.

116. (Previously Presented) The computer program product of claim 113, wherein said set of instructions is further configured to:
if said request does not require said second TCP connection with one of said plurality of servers,
service said request for data, and
close said connection with said client.

117. (Original) The computer program product of claim 116, wherein said request for data is served by passing data from said data switching unit to said control unit for transmission to said application on said client.

118. (Original) The computer program product of claim 116, wherein said set of instructions is further configured to:
if said request requires said second TCP connection with one of said plurality of servers,
select a first server from said plurality of servers, and
initiate said second TCP connection with said first server.

119. (Original) The computer program product of claim 118, wherein said application requests said end-to-end TCP connection with said first server.

120. (Previously Presented) The computer program product of claim 118, wherein said set of instructions is further configured to:
receive said data on said second TCP connection from said first server;
store said data in said receive buffer of said second TCP connection.

121. (Previously Presented) The computer program product of claim 120, wherein said set of instructions is further configured to:
if said first TCP connection needs said additional data,
request said additional data from said first server.
122. (Original) The computer program product of claim 121, wherein said additional data is transferred into said transmit buffer without a request for said additional data.

123. **(Currently Amended)** The computer program product of claim 121, wherein said set of instructions is further configured to:
if said request for data from said application is served,
close said first TCP connection with said client **in response to receiving a request for closing said connection from said data switching unit**.

124. **(Currently Amended)** The computer program product of claim 123, ~~**wherein said closing of said connection is done by said control unit upon a**~~

~~receiving a request for closing said connection from said data switching unit~~ wherein said set of instructions is further configured to perform a two-stage operation for releasing a control memory entry for said connection, wherein said control memory entry comprises a connection block entry and a flow control entry, and said two-stage operation comprises:

releasing said connection block entry prior to receiving said request for closing said connection; and

releasing said flow control entry in response to receiving said request for closing said connection.